



COMILLAS

UNIVERSIDAD PONTIFICIA

ICAI

ICADE

CIHS

Proyecto Final

Radar Velocidad

Jaime Pedrosa Comino
Ignacio Viadero Canduela
Grupo B

Arquitectura Big Data
3º Grado en Ingeniería Matemática e Inteligencia Artificial

Índice

INTRODUCCIÓN	3
METODOLOGÍA	4
RESULTADOS.....	7
CONCLUSIÓN.....	8

Introducción

En este informe vamos a contar cómo desarrollamos un sistema de radar automático que, usando una cámara y un poco de código, es capaz de detectar vehículos y extraer la velocidad en tiempo real a la que se dirigen. La idea es que cuando un coche pase frente a la cámara, se procese la información mediante lo aprendido a lo largo de la asignatura para poder extraer la velocidad de cada coche detectado

La seguridad vial es un tema importante. Un sistema como este puede ayudar a evitar accidentes, ya que detecta infracciones sin necesidad de intervención humana. Además, es un proyecto que combina tecnología y creatividad, lo que lo hace interesante y útil.

Contexto

En este caso, hemos decidido realizar el proyecto a menor escala, ya que se nos proporcionaba la cámara de la raspberry, hemos decidido procesar y extraer la velocidad de un coche del juego infantil Scalextric, donde tenemos un coche dando vueltas a un circuito. Asimismo, hemos creado un sistema que haga lo necesario para realizar correctamente nuestro radar: observar, calcular la velocidad y mostrar por pantalla la velocidad actual de cada coche.

Objetivos

- Desarrollar un radar automático que detecte y calcule la velocidad de los vehículos en tiempo real.
- Generar una alerta cada vez que se detecte un vehículo que y muestre su velocidad.
- Asegurarnos de que el sistema funcione de manera rápida y precisa.

GitHub

El link para el repositorio de github donde encontramos todos los ficheros de código es:
<https://github.com/IgnacioViadero/ComputerVision>

Metodología

Descripción del Entorno de Desarrollo

Para llevar a cabo este proyecto, utilizamos las siguientes herramientas:

- **Lenguaje de Programación:** Python
- **Librerías:**
 - OpenCV: para procesar imágenes y detectar vehículos.
 - NumPy: para realizar cálculos matemáticos.
 - Math: para medir distancias.
- **Hardware:**
 - Raspberry Pi 4 con una cámara Módulo 3 WIDE.

El entorno de desarrollo fue una Raspberry Pi, que nos permitió capturar video y procesarlo en tiempo real. Controlamos la Raspberry Pi de forma remota usando SSH y escritorio remoto. Se realizó una previa calibración de la cámara, la cual a lo largo de la elaboración del proyecto nos dio fallos, y no pudimos utilizar la cámara para la grabación de nuestro circuito con coches, por lo que decidimos utilizar otro dispositivo.

Diseño de la Solución

El sistema se dividió en tres partes:

1. **Captura de Video:**
 - La cámara de la Raspberry Pi graba en tiempo real.
 - Los fotogramas capturados se convierten en imágenes para analizarlas.
2. **Rastreo de Vehículos y Cálculo de Velocidad:**
 - Implementado utilizando el archivo auxiliar rastreador.py.
 - Se divide en 3 zonas el rastreo y detección de vehículos.
 1. Una primera zona, donde se realiza la detección de los vehículos utilizando los métodos vistos en clase y procedentes de la librería cv2 de detección de contornos. En un primer lugar, nos dio fallos debido a que no se reconocían correctamente todos los vehículos, esto era debido a que el umbral mínimo para considerarlo coche, en este caso el área de píxeles de cada contorno era muy elevado, por ello decidimos bajarlo a 1000, donde ahora si que detecta correctamente cada vehículo. Cabe destacar que es necesario variar este umbral dependiendo del tipo de vídeo que tenemos, debido a que el tamaño de cada coche que se muestra en los frames puede variar dependiendo de la posición de la cámara.
 2. Una segunda zona, la cual realiza el cálculo de la velocidad. Esta detecta un vehículo que acaba de entrar en el área 2, y realiza un conteo de frames de cuanto tarda en ejecutarse hasta que sale del área. Este número de frames había que pasarlo a segundos, y como en nuestro caso nuestro dispositivo grababa en 30 fps, hacíamos la relación y Nosotros previamente realizamos la medida física de cuantos centímetros correspondían a la zona 2 en nuestro circuito, y luego la posterior división para calcular la velocidad dividiendo la distancia por el tiempo.
 3. Finalmente, una tercera zona, donde se muestra por pantalla en una pestaña encima de cada coche, en la que se muestra la velocidad actual de cada vehículo.

Secuencia de Transformación de la Imagen

1. **Captura de Fotogramas:** Se captura el video desde la cámara en tiempo real.
2. **Conversión a Escala de Grises:** Se convierten los fotogramas a imágenes en escala de grises para facilitar el procesamiento.
3. **Detección de Bordes:** Se aplican algoritmos de detección de bordes para identificar vehículos en la imagen.
4. **Rastreo de Objetos:** Se identifican y siguen los vehículos en fotogramas consecutivos.
5. **Cálculo de Velocidad:** Se calcula la velocidad en función de la distancia recorrida por los vehículos y el tiempo transcurrido.

Pruebas Realizadas

Se realizaron pruebas exhaustivas en diferentes escenarios para garantizar la precisión y fiabilidad del sistema.

Las pruebas incluyeron:

- Videos simulados con diferentes condiciones de iluminación y ángulos de cámara.
- Variación de velocidad de los vehículos para comprobar el correcto funcionamiento del radar y del detector de contornos ante diferentes velocidades

Durante las pruebas, se ajustaron los parámetros de detección y cálculo de velocidad para optimizar el rendimiento del sistema. Se comprobó que el sistema podía procesar múltiples vehículos simultáneamente sin comprometer la tasa de refresco.

Resultados

Descripción de los Resultados

El sistema fue capaz de rastrear vehículos en movimiento y calcular su velocidad con una precisión muy alta. En los escenarios de prueba, se detectaron y etiquetaron correctamente los vehículos con el muestreo en tiempo real de su velocidad

Discusión de los Resultados

El sistema funcionó como esperábamos. Detectó correctamente todos los vehículos, incluso tanto aquellos que iban demasiado rápido como los que iban más lento y mostró perfectamente sus velocidades. Sin embargo, hubo algunos retos:

- La iluminación variable afectaba la precisión.
- Cambiar la posición de la cámara alteraba los resultados, por lo que era importante mantener una configuración constante.
- Cambiar la posición de la cámara afectaba en el tamaño de los vehículos, lo que variaba la detección de contornos. Por ello, se decidió un sitio en concreto donde posicionar la cámara y no se varió en todo el proyecto.

Conclusión

Resumen del Proceso

El desarrollo del sistema comenzó con la configuración del entorno de desarrollo y la calibración de la cámara. A partir de ahí, se implementaron los módulos de rastreo de objetos y cálculo de velocidad. Finalmente, se llevaron a cabo pruebas para validar el sistema y ajustar los parámetros según los resultados obtenidos.

Principales Logros

- Implementación exitosa de un sistema de radar automático para la detección de velocidad.
- Alta precisión en la detección y cálculo de velocidad, alcanzando una alta precisión.
- Funcionamiento correcto de extracción de velocidad en tiempo real en un entorno real pero a menor escala (circuito de coches).

El proyecto demuestra la viabilidad de utilizar visión por ordenador para poder detectar vehículos y procesar sus velocidades mediante su detección automático. Las futuras ampliaciones podrían incluir el uso de modelos más avanzados de detección de objetos y la integración con sistemas de gestión de tráfico.